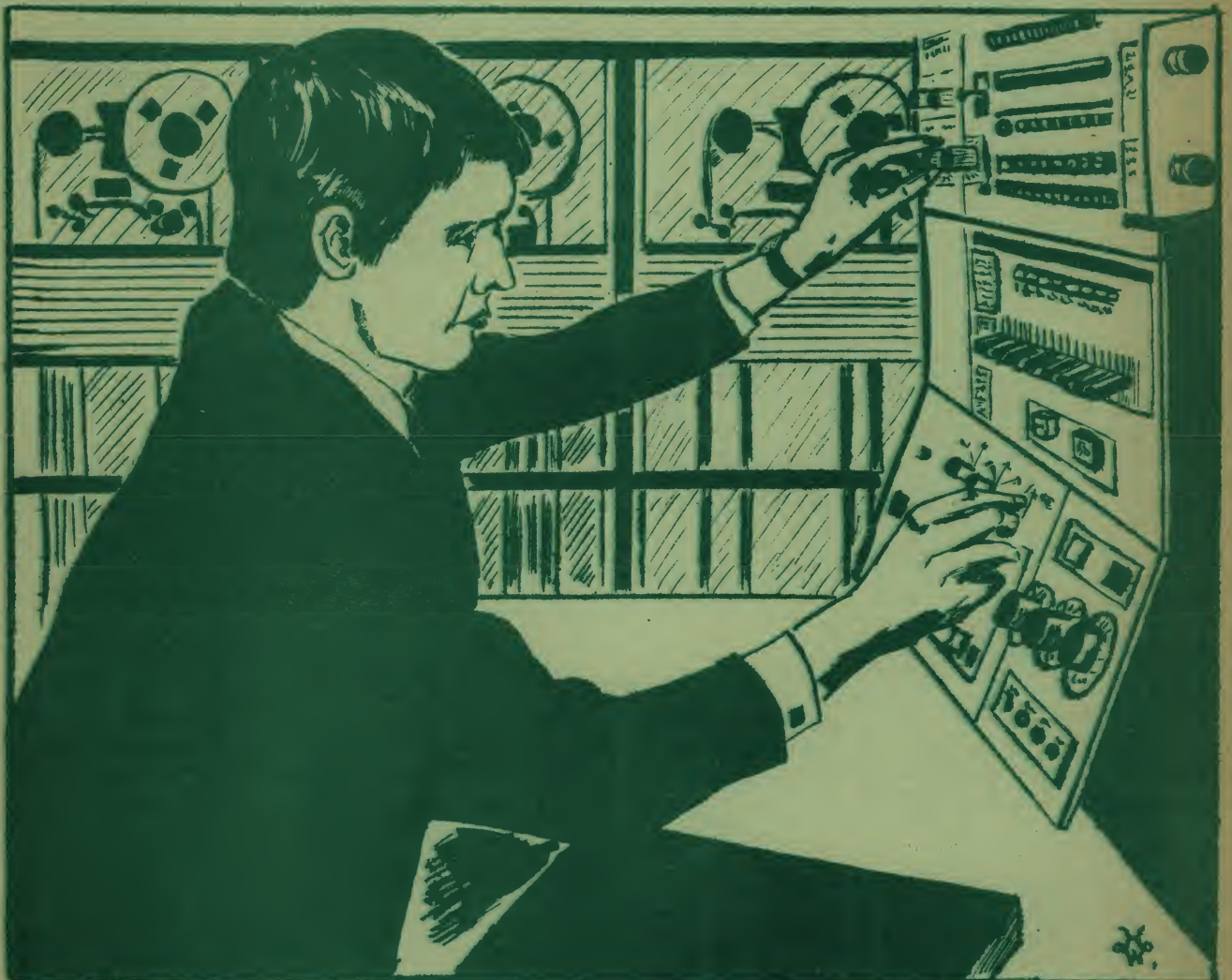


COMPUTER PROGRAMMEUR



MAGNEETBANDEN

Ir. P.A. Tas

Een publikatie van de Stichting Het Nederlands Studiecentrum voor Informatica, Amsterdam.

Copyright © Studiecentrum voor Informatica, 1971.
Niets uit deze uitgave mag worden vermenigvuldigd en/of openbaar gemaakt door middel van druk, fotokopie, microfilm of op welke andere wijze dan ook zonder voorafgaande schriftelijke toestemming van de uitgever.

1. INLEIDING

Zoals door het voorafgaande wel duidelijk geworden is, zijn de rekenkundige bewerkingen van een programma ter verwerking van administratieve gegevens in het algemeen zeer eenvoudig. Dit betekent meestal dat de tijd die dit soort bewerkingen neemt, zeer kort is, en dat de tijdsduur van een probleem in hoofdzaak bepaald wordt door de snelheid van in- en uitvoer van de gegevens.

Een ponskaartlezer leest gemiddels 500 kaarten per minuut, dus 1 kaart per $60/5000 = 0,12$ sekonde.

Een snelle machine heeft een gemiddelde bewerkingstijd voor een opdracht van 40 mikrosekonden = $40/1000000$ sekonden = $40 \cdot 10^{-6}$ sekonden.

In de tijd dat een kaart gelezen wordt zouden dan $120/40 \times 10^3$, dat is ongeveer $3 \times 10^3 = 3000$ instructies uitgevoerd kunnen worden.

Een ponskaartmachine pons max. 250 kaarten per minuut, hetgeen betekent dat in de tijd dat een kaart geponsd wordt er 6000 instructies uitgevoerd kunnen worden.

Uit bovenstaande blijkt wel, dat in- en uitvoer ongunstig afsteekt t.o.v. de bewerkingstijd van opdrachten. In vele gevallen zal de machine meer wachten dan werken. Er zijn een aantal mogelijkheden om meer doelmatig te werken. Eén daarvan ligt natuurlijk voor de hand, n.l. het opvoeren van de snelheid van in- en uitvoer. In plaats van kaarten kan bijv. gebruik gemaakt worden van MAGNEETBANDEN.

PARALLELARBEID

Een andere mogelijkheid is een systeem van tegelijkertijd rekenen en in- en uitvoeren. Een groep van gegevens die in het geheugen staan worden verwerkt, terwijl een nieuwe groep gegevens tegelijkertijd ergens in het geheugen gebracht of uitgevoerd wordt. Dit wordt wel *parallelarbeid* genoemd, een slechte benaming omdat het de indruk wekt dat tegelijkertijd gerekend wordt door 2 verschillende programma's hetgeen met één rekenorgaan onmogelijk is. Het beste is natuurlijk de beide genoemde mogelijkheden te combineren.

Bijvoorbeeld: Invoer van en uitvoer op magneetband terwijl tijdens in- en uitvoer ook gerekend kan worden. In dat geval moet eerst de informatie van ponskaarten of ponsbanden op magneetbanden gebracht worden. Dit geschiedt m.b.v. de rekenmachine.

Een van de meest efficiënte, maar ook dure oplossingen van

het in- en uitvoer probleem wordt gegeven door het gebruik maken van twee rekenmachines. Een kleine machine die informatie van ponsband of ponskaarten op een magneetband brengt. Een grote machine die de magneetbanden verwerkt. De uitvoer komt dan ook op een magneetband die dan eventueel via de kleine machine zichtbaar gemaakt kan worden op een sneldrukker.

Bij de moderne machines worden een aantal programma's tegelijkertijd in het geheugen opgeslagen. Een coördinatieprogramma zorgt er voor dat deze programma's parallel kunnen werken.

Enkele van de programma's zouden nu de taak van de kleine machine kunnen overnemen, zodat de konversie van kaarten of ponsband naar magneetband en omgekeerd van magneetband naar sneldrukker of ponskaarten ook door de grote machine wordt verricht, maar dan terwijl tegelijkertijd nog andere programma's actief zijn.

HOEVEELHEID INFORMATIE SPEELT EEN ROL

Een geheel ander aspekt van de administratie is de hoeveelheid informatie. In het algemeen zullen bijzonder veel gegevens verwerkt moeten worden hetgeen veel kaarten of grote rollen ponsband vereist. Veel beter is dan gebruik te maken van magneetbanden die bijzonder veel informatie kunnen bevatten en bovendien vrij eenvoudig te hanteren zijn. De magneetband wordt hier dan gebruikt als opslagplaats van gegevens.

MAGNEETBAND ALS SNELGEHEUGEN

Een andere toepassing van de magneetband is om deze te gebruiken als aanvulling op het snelgeheugen. Tijdens de afloop van een programma kunnen tijdelijk tussenresultaten naar de band gebracht worden. Dit wordt alleen gedaan als het snelle geheugen niet toereikend is.

Een programma dat in zijn geheel niet in het geheugen past, kan eerst op de magneetband gebracht worden.

Nu wordt een gedeelte van het programma van de band in het geheugen gelezen en gestart. Tegen de tijd dat het volgende stuk programma, dat nog op de band staat, aan de beurt is, wordt dit ingelezen en overschrijft dan een gedeelte van het eerste stuk, etc.

2.0 BESTANDEN

In de administratie is het vaak noodzakelijk om een groot aantal gegevens op te slaan. Deze moeten op een ordelijke manier gerangschikt zijn opdat ze snel bereikt kunnen worden.

Er zijn allerlei soorten activiteiten mogelijk met deze gegevens, zoals informatie betrekken, ingewikkelde statische

berekeningen, veranderen van gegevens, etc.

Een bestand is nu een verzameling van *RECORDS*. Elk record bestaat uit een aantal gegevens. Deze gegevens zijn in elk record van dezelfde aard en bevatten informatie over mensen, artikelen etc. Een personeelsadministratie van een groot bedrijf, zal van elke werknemer een groot aantal gegevens verzamelen in een record. Elk record bevat dezelfde soort gegevens zoals : afdeling, registratie, nummer, naam, etc. Het totaal aantal records wordt samengevat in een bestand. Het is duidelijk dat er een volgorde moet zijn, waarin die records in het bestand voorkomen.

DE SLEUTEL

Elk record heeft dan ook een identificatienummer, *DE SLEUTEL*, die de plaats in het bestand bepaalt.

Een record kan meerdere sleutels bevatten afhankelijk van de wijze waarop het bestand georganiseerd is. Hiermee wordt bedoeld dat een bestand bijv. op volgorde kan liggen wat betreft het registratienummer, maar hetzelfde bestand kan men ook alfabetisch op naam rangschikken.

In een bedrijf wil men permanent op de hoogte zijn van bepaalde aspecten, zoals voorraden, omzetten, etc. Deze gegevens worden dan ook in een bestand opgeslagen en er worden methoden ontworpen om snel informatie te kunnen opvragen. Aangezien deze bestanden natuurlijk zoveel mogelijk "up to date" moeten zijn zullen zij vaak veranderd moeten worden. De veranderingen worden verzameld in mutatie-bestanden.

Mutatie betekent verandering. Met behulp van de mutatie-bestanden, worden de informatie-bestanden gewijzigd.

VOORBEELD

Nemen we als voorbeeld een voorraad-bestand, waarbij per artikelnummer een record bestaat met o.a. het aantal artikelen in voorraad. We zullen dit het **hoofdbestand** noemen.

De voorraad zal veranderen en dit zal op de een of andere manier aan het voorraad-bestand bekend gemaakt moeten worden. Er wordt nu een bestand gemaakt waarin records met een art. nummer en een getal dat de veranderingen van de voorraad aangeeft. Dit laatste bestand (*mutatie-bestand*) bevat nu alleen die artikelnummers van het hoofdbestand, waarvan de voorraad veranderd is.

We zullen ons hier beperken tot bestanden die opgeslagen zijn op magneetbanden of ponskaarten. Het is n.l. heel goed mogelijk en vaak ook bijzonder wenselijk een bestand op een schijvengeheugen of een ander random access geheugen te brengen.

Aangezien deze geheugens in de SERA niet programmeerbaar zijn, wordt dat soort bestanden hier niet behandeld. Indien de inhoud van een bestand veranderd wordt spreekt men van activiteit.

Als er gezegd wordt dat het bestand voor 80% actief is, dan wordt bedoeld dat 80% van de records betrokken zijn bij veranderingen. Een bestand dat te maken heeft met salarissen heeft in het algemeen hoge activiteit. Een voorraadbestand kan een lage activiteit bezitten. Indien we bij zo'n bestand spreken van 5% activiteit dan wil dat zeggen dat 5% van de artikelen een voorraadsverandering hebben ondergaan.

DE SPLITSING VAN EEN ACTIVITEIT

In het algemeen kan men zeggen dat activiteit te splitsen is in drie groepen:

- a. vergroting van het bestand*
- b. verandering in het bestand*
- c. verkleining van het bestand.*

De bestanden op magneetbanden worden *SERIE-BESTANDEN* genoemd.

De serie-bestanden zijn tot nu toe nog het meest in gebruik. De records worden de een na de ander verwerkt en zijn altijd volgens de een of andere sleutel in volgorde gebracht.

Drie zeer belangrijke karakteristieken van een in serie georganiseerd bestand zijn :

- 1. Alle activiteiten moeten in volgorde worden gebracht, voordat er met het hoofd-bestand gewerkt gaat worden.*
- 2. Alle informatie in het hoofd-bestand moet worden bekeken, hoewel in veel toepassingen een gedeelte ervan niet wordt gebruikt.*
- 3. Zoveel mogelijk "aanvragen" moeten verzameld worden, opdat zo efficiënt mogelijk gebruik gemaakt kan worden van een doorgang in het bestand.*

We zullen ons in het hier volgende bezig houden met hoofdbestanden die opgeslagen zijn op magneetbanden.

HOE MOET ER NU MET MAGNEETBAND GEWERKT WORDEN?

Er bestaan opdrachten die het volgende kunnen doen:

- a. lees een blok van de magneetband en plaats de woorden uit dit blok achtereenvolgens in het geheugen, te beginnen bij adres n. Met de leeskop van het magneetbandapparaat wordt het eerste woord uit het blok gelezen van de*

Ir. P.A. Tas

magneetband en weggebracht in het geheugen naar adres n . Daarna wordt het volgende woord gelezen en weggebracht naar adres $n+1$, dan het daarop volgende woord naar adres $n+2$ etc.

Totdat een blokhiaat ontmoet wordt.

Zoals bekend wordt informatie op de magneetband geschreven in blokken van woorden. Deze blokken worden gescheiden door blokhiaten. Als er dus van een magneetband gelezen wordt dan beweegt de band zich altijd over een vol blok tot het volgende blokhiaat.

- b. Schrijf vanuit geheugen te beginnen met adres n achtereenvolgens een aantal woorden (een blok) op de magneetband. De magneetband wordt op gang gebracht en daarna wordt het woord uit adres n , $n+1$, etc. op de band geschreven totdat het aantal opgegeven woorden bereikt is, dan wordt het blokhiaat geschreven en de magneetband stopt.

Op deze wijze wordt informatie van en naar het geheugen getransporteerd. De magneetband is gevuld met blokken woorden, gescheiden door blokhiaten.

In een bestand, zou bijv. elk record net een blok kunnen beslaan van de magneetband.

Er wordt hier wel op gewezen dat dit beslist niet noodzakelijk is, er kunnen meerdere records in een blok opgeslagen worden. Een record is n.l. een eenheid van de magneetband. Deze eenheden behoeven zeker niet dezelfde te zijn.

DE MAGNEETBAND HEEFT EEN AANTAL TECHNISCHE BEPERKINGEN

Een van de belangrijkste is wel dat het onmogelijk is om nog informatie te lezen na het punt waarop de schrijfkop heeft opgehouden te schrijven.

De consequentie hiervan is dat bij het veranderen van een bestand altijd het veranderde bestand op een nieuwe magneetband geschreven zal moeten worden.

Records die niet aan verandering onderhevig zijn worden ingelezen vanaf de band met het oude bestand en onveranderd op de band met het nieuwe bestand geschreven.

Elke record dat wel veranderd moet worden, wordt vanaf de oude bestandband ingelezen, veranderd en geschreven op de nieuwe bestandband. Een bestandsverandering betekent dus altijd minstens twee magneetbanden. Eén met het oude bestand, één met het nieuwe bestand.

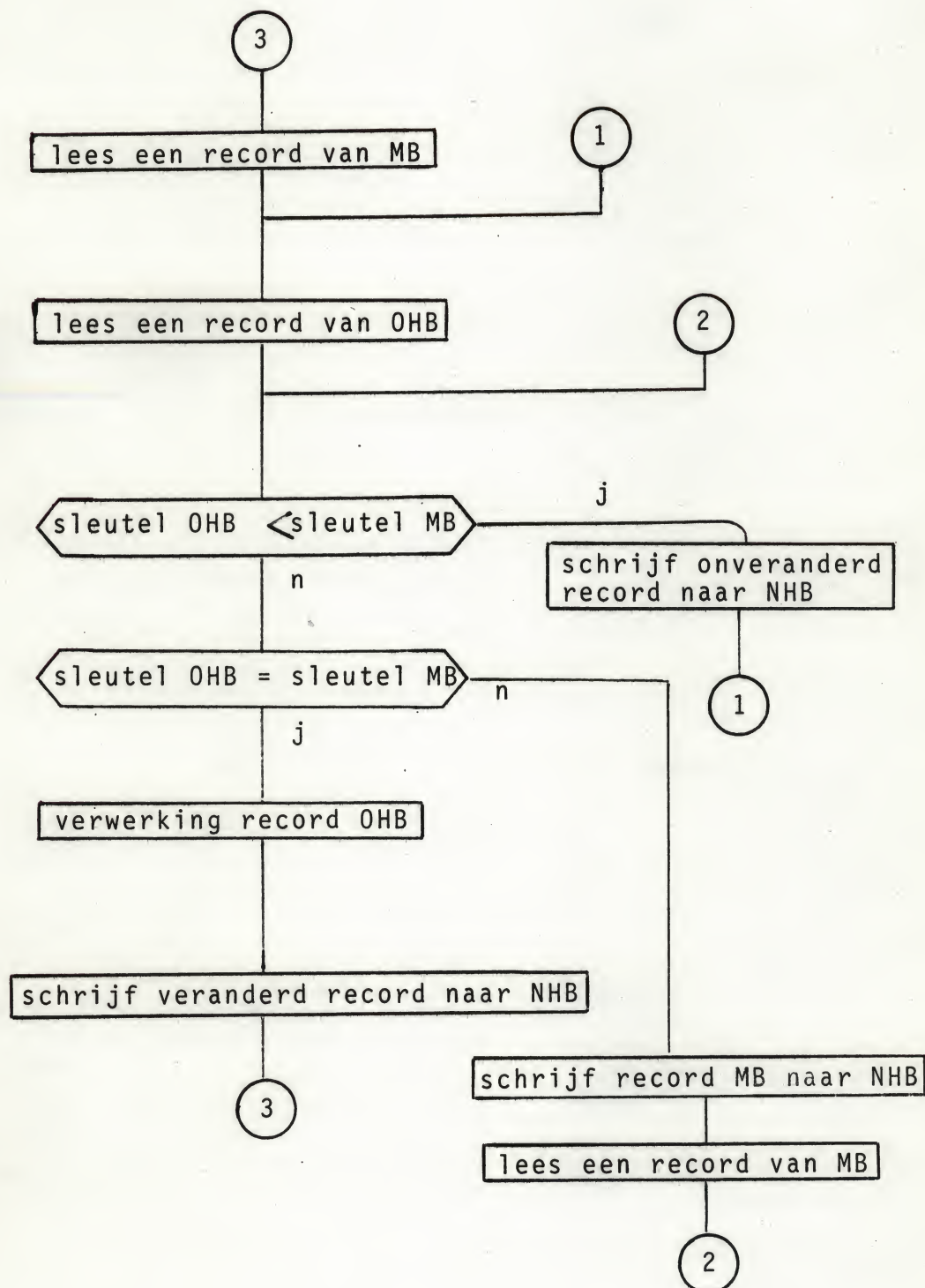
2.1 BESTANDVERWERKING

Het volgende stroomdiagram geeft een voorbeeld van een bestandsverwerking. De volgende afkortingen zijn gebruikt :

OHB = oude hoofd-bestand

NHB = nieuwe hoofd-bestand

MB = mutatie-bestand



Ir. P.A. Tas

We hebben hier zeer in het algemeen een indruk van de verwerking van een serie-bestand.

Er zijn vele belangrijke dingen echter niet besproken.

- 1e. Door foute behandeling behoeft een bestand niet altijd in volgorde te zijn. Het is van belang dit te testen en indien noodzakelijk correcties aan te brengen.
- 2e. We zijn er hier van uit gegaan dat het mutatie-bestand geen twee records heeft met dezelfde sleutel. Dat zal echter vaak voorkomen; de klant kan bijv. twee aankopen gedaan hebben.
Het stroomdiagram zou moeten worden uitgebreid om deze mogelijkheid te kunnen verwerken.
- 3e. Er moeten testen worden ingebouwd om te voorkomen dat een foute mutatie-code informatie verstoort. Bijv. door een foute ponsing in de mutatie-code wordt een record weggelaten inplaats van veranderd, of een nieuwe klant krijgt een nummer van een reeds bestaande klant, etc.
- 4e. Er moet getest kunnen worden of het proces klaar is. Op de een of andere manier moet de machine kunnen ontdekken wanneer het einde van het bestand is bereikt. Dit wordt vaak gedaan door toepassing van sluitblokken.

VOORBEELD

Een voorraad-bestand van ongeveer 50.000 artikelen staat op een magneetband.

Elk record van het bestand is 10 woorden groot en bevat in het eerste woord een artikelnummer en in het tweede woord het aantal aanwezige artikelen in voorraad. De laatste 8 woorden doen hier niet terzake.

Elk record beslaat een blok op magneetband. De records zijn op volgorde van artikelnummer gerangschikt.

Het bestand op de band wordt afgesloten door een blok van 10 woorden.

Het eerste woord van dit blok is gevuld met 8 hexaden z.

Dit voorraad-bestand moet worden bijgewerkt. Dit wordt gedaan met behulp van kaarten.

Gegeven: een pak kaarten met sluitkaart. Op elke kaart staat in kolom 10 t/m 14 een artikelnummer.

In kolom 5 staat de mutatie-code:

0 verandering

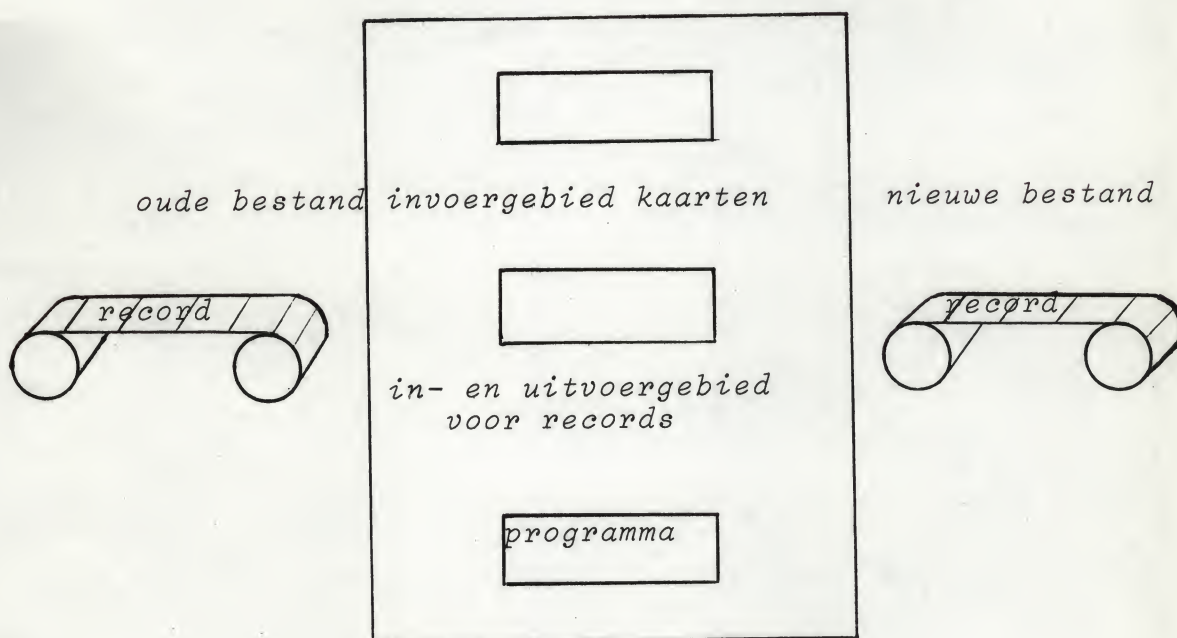
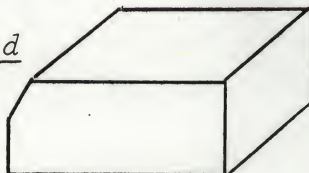
1 weglaten van het betreffende record.

Tussenvoegen treedt hier dus niet op. In het geval van een verandering in de voorraad staat in kolom 30 t/m 36 een getal met teken die de verandering aangeeft. Een negatief ge-

tal geeft aan dat de voorraad verminderd is.

De kaarten zijn op volgorde van artikelnummer gebracht.

kaarten mutatie bestand



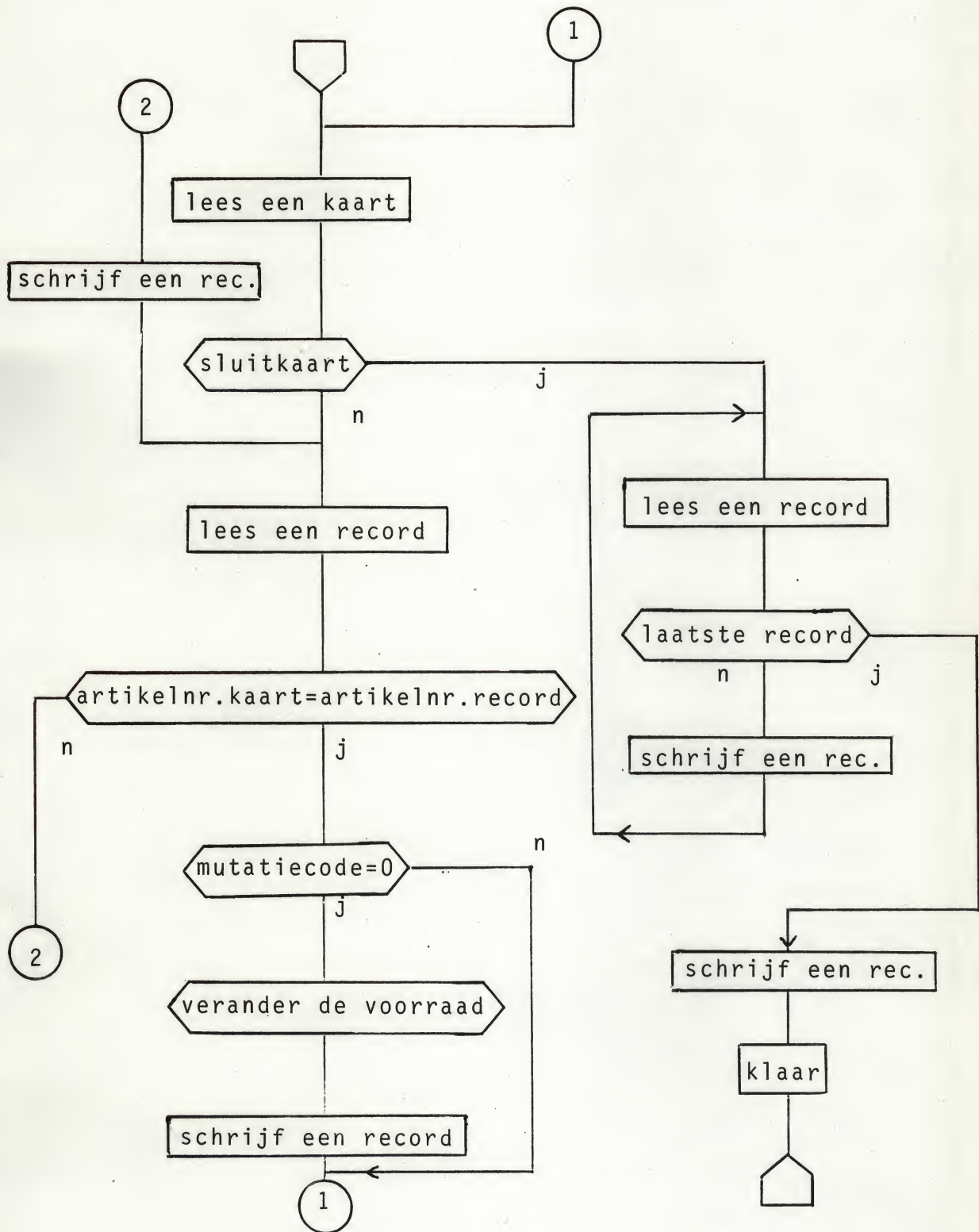
Bovenstaand figuur geeft een schematisch overzicht van de organisatie.

Er is een invoergebied voor kaarten waarin telkens een kaart gelezen wordt.

In het geval van de SERA is dit de buffer.

Er is een invoergebied voor een record van het oude bestand. Ditzelfde invoergebied wordt weer gebruikt als uitvoergebied om een record weg te schrijven naar het nieuwe bestand. Dit in- en uitvoergebied heeft de lengte van een blok op de magneetband.

Onderstaand stroomdiagram geeft een oplossing van het probleem. Er is volstaan met de tekst : schrijf een record en lees een record, omdat het duidelijk is waarvan gelezen en waarop geschreven wordt.



Ir. P.A. Tas

3. MAGNEETOPDRACHTEN EN PARALLELWERKING

3.0 INLEIDING

Voor het besturen van een magneetbandapparaat zijn een aantal opdrachten aanwezig die het magneetbandapparaat doen starten.

Verder moet dit apparaat weten of het in- of uitvoer betreft en wat de bewegingsrichting zal zijn van de band.

Ontmoet het besturingsorgaan zo'n magneetbandopdracht, dan wordt deze onmiddellijk uitgevoerd.

Nu is in het begin van dit hoofdstuk gesproken over *parallelwerking*. Hiermee werd bedoeld : tegelijkertijd rekenen en in- of uitvoeren.

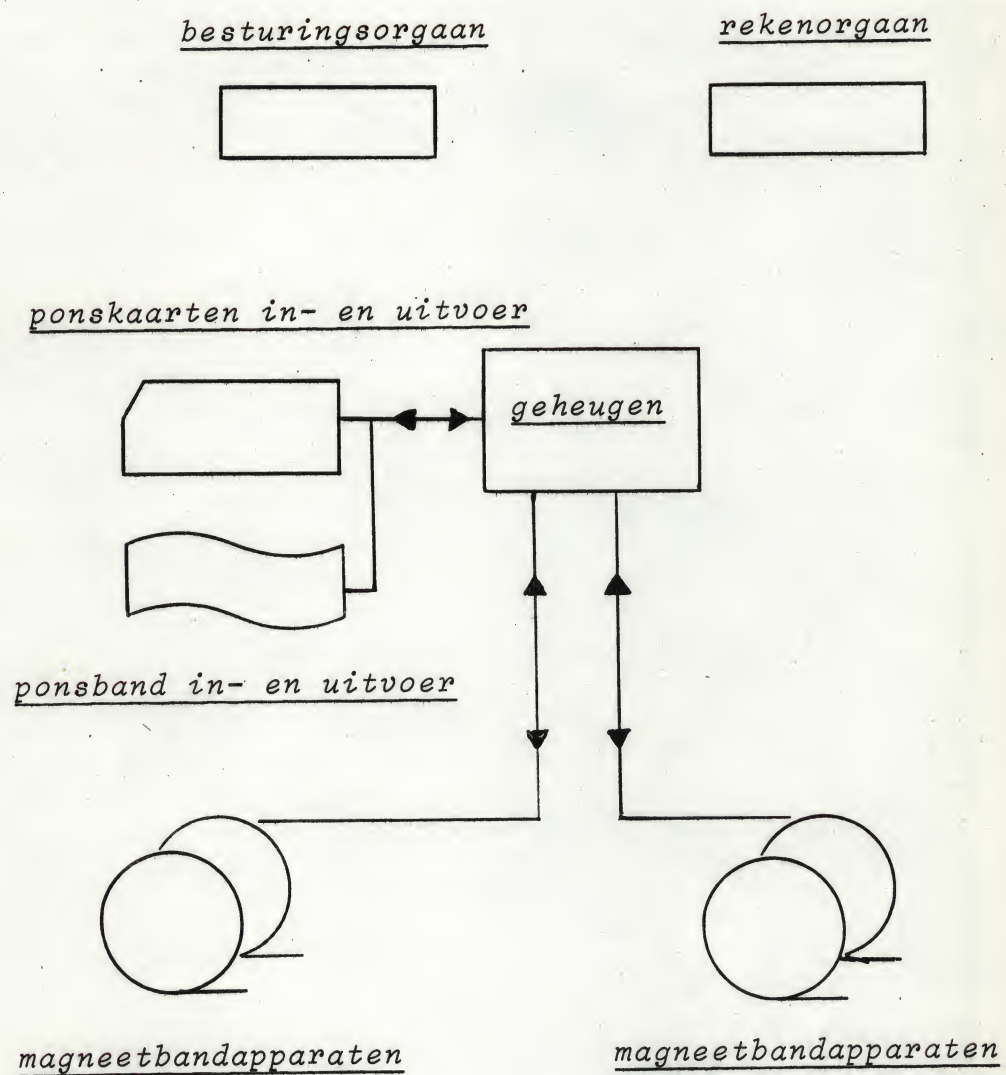
De SERA heeft wat betreft de magneetbanden, deze faciliteiten ook. Het zal dus mogelijk blijken te zijn om met een bepaald stuk geheugen te rekenen, terwijl tegelijkertijd een ander stuk met informatie gevuld wordt.
Dit dan voor de invoer.

Voor de uitvoer zal natuurlijk iets dergelijks gelden. Wel wordt de programmering iets moeilijker, ondanks een coördinatorprogramma die reeds een groot deel van de moeilijkheden opvangt.

Dit coördinatorprogramma, MagneetBand Lees en Schrijfprogramma genaamd, en afgekort MBLS, gaat de in- en uitvoer regelen en maakt gebruik van de echte startopdrachten voor de magneetband.

Deze opdrachten mogen door de programmeur niet gebruikt worden; daarvoor in de plaats krijgt deze de beschikking over zgn. pseudostartopdrachten.

3.1 ALLEREERST EEN OVERZICHT VAN DE CONFIGURATIE VAN DE SERA MACHINE

Figuur 1.DE BUFFER

Aan het geheugen zijn ponskaarten in- en uitvoer en ponsband in- en uitvoer gekoppeld. Deze apparaten zijn via een kanaal met een speciaal stuk van het geheugen verbonden : de BUFFER.

Alle in- en uitvoer met ponskaarten en ponsband loopt via de BUFFER. Verder kunnen aan dat geheugen magneetbandapparaten gekoppeld worden.

Deze koppeling geschiedt via speciale in- en uitvoerkanalen.

Ir. P.A. Tas

Deze kanalen kunnen onafhankelijk van elkaar informatie overdragen van het geheugen van de machine naar magneetbanden en omgekeerd.

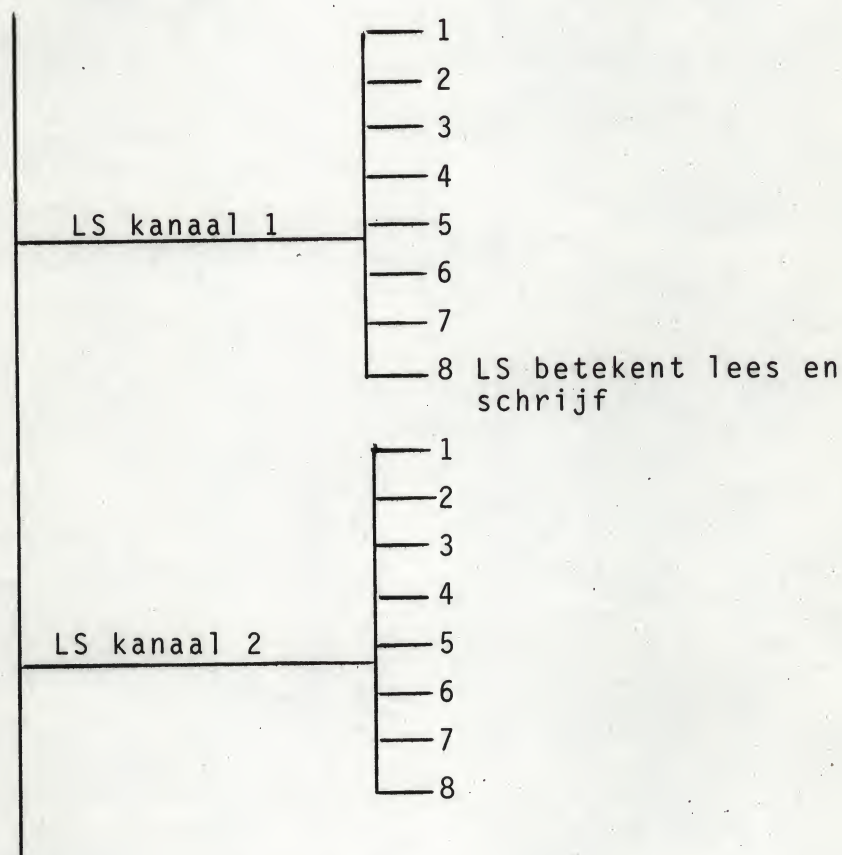
Dit geheugen is zeker niet beperkt tot de buffer.

Men kan informatie van de magneetband naar elk willekeurig stuk geheugen overbrengen en omgekeerd.

3.2. KANALEN

Het is mogelijk om tegelijkertijd informatie over beide kanalen te transporteren. Er zijn er in fig. 1 twee kanalen getekend, maar dit aantal kan worden uitgebreid.

Bekijken we de kanalen iets nauwkeuriger dan ziet het er uit als in fig. 2.



Figuur 2.

Aan elk kanaal kan een aantal (max.8) magneetbandapparaten aangesloten worden die echter niet onafhankelijk van elkaar kunnen werken.

Ir. P.A. Tas

Deze magneetband apparaten worden genummerd van 1 t/m 8.
De kanalen worden ook genummerd.

Op deze wijze kunnen we spreken over magneetbandapparaat 1.1, d.w.z. magneetbandapparaat 1 op kanaal 1 of magneetbandapparaat 2.6, d.w.z. magneetbandapparaat 6 op kanaal 2. Aangezien het mogelijk is om tegelijkertijd informatie te transporteren over de kanalen kan bijv. een blok gelezen worden van magneetbandapparaat 1.4, terwijl tegelijkertijd een blok geschreven wordt op magneetbandapparaat 2.3.

Het is niet mogelijk om tegelijkertijd bijvoorbeeld 2 magneetbanden te laten werken die op hetzelfde kanaal zijn aangesloten.

3.3 PARALLELWERKING

Bij de ponskaart- en ponsbandopdrachten wordt met uitvoering van de daarop volgende opdrachten gewacht totdat de in- of uitvoer is geschied. Indien dit ook het geval zou zijn bij de magneetbandopdrachten dan zouden er nooit twee magneetbandapparaten tegelijk kunnen werken al zouden ze op verschillende kanalen zijn aangebracht. Bovendien zou parallelwerking onmogelijk zijn. Magneetbandopdrachten verschillen daarom van andere in- of uitvoeropdrachten. Het essentiële van deze opdrachten zal moeten zijn dat door middel van een magneetbandinstructie direkt na het starten van het magneetbandapparaat de volgende opdrachten van het programma zullen worden uitgevoerd. Er ontstaat nu de situatie dat het magneetbandapparaat gestart is en daarna een blok informatie verplaatst zal worden van of naar de magneetband, maar dat tegelijkertijd met die verplaatsing de instructies volgend op de magneetbandinstructie uitgevoerd zullen worden.

Er wordt nu dus niet gewacht door het programma op het einde van het in- of uitvoeren van een blok informatie. De magneetbandopdracht is veranderd in een startopdracht voor het magneetbandapparaat. Het gevolg van deze startopdracht zoals verplaatsing van een blok informatie van en naar de magneetband, gaat verder geheel onafhankelijk en gelijktijdig met het uitvoeren van verdere opdrachten van het programma.

In figuur 3 is een programma-opdracht geschetst waarbij op een bepaald ogenblik een startopdracht voor een magneetband komt om een blok van de band in het geheugen te lezen. Deze opdracht zal nu een magneetband starten die dan woorden gaat verplaatsen van de band naar het geheugen. Na het geven van de startopdracht gaat het besturingsorgaan direkt verder met de volgende opdracht van het programma. Tegelijkertijd worden nu dus de opdrachten van het programma

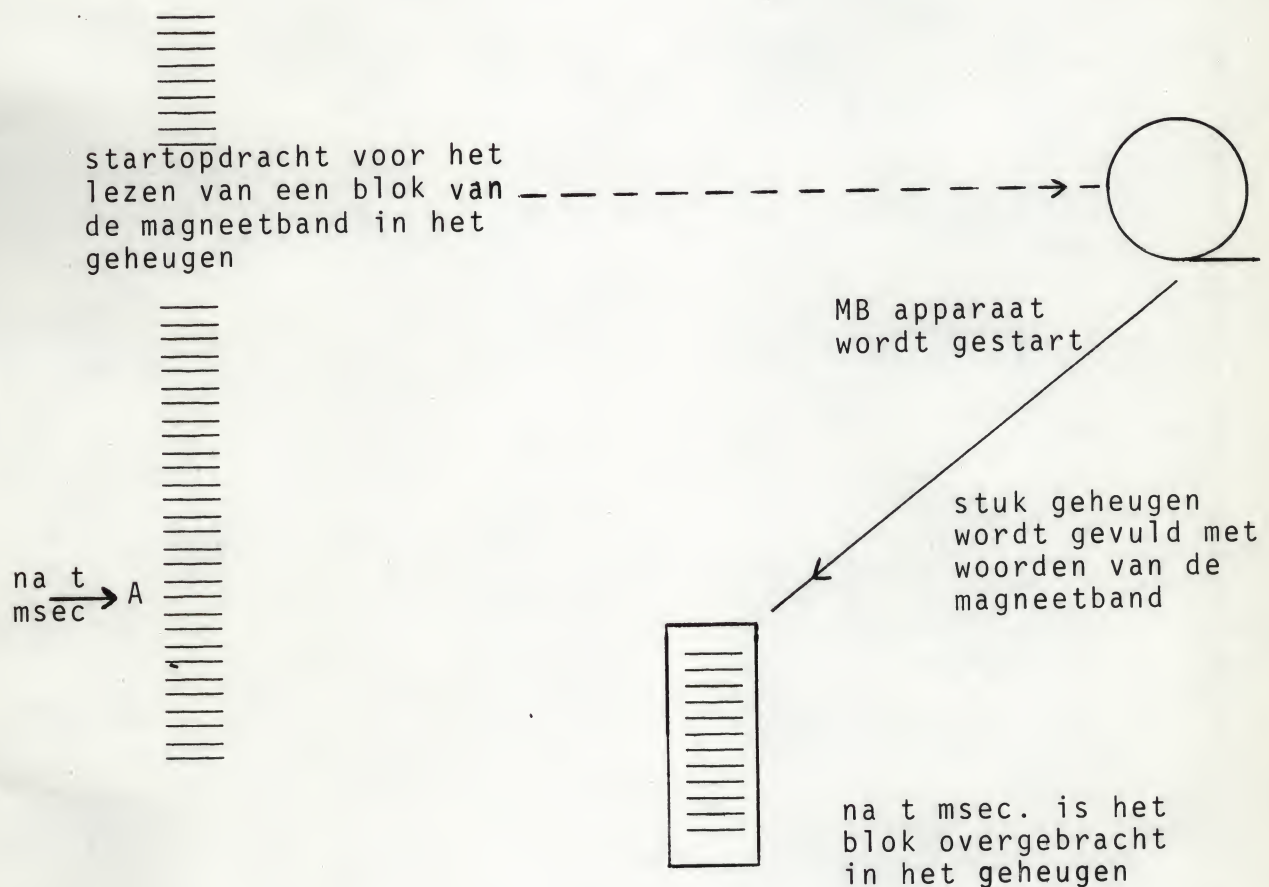
Ir. P.A. Tas

uitgevoerd en vindt er een informatie-overdracht plaats van de band naar het geheugen.

Na t msec. zal het programma punt A bereiken hebben en zal het blok informatie verplaatst zijn.

Het parallel werken betreft dus zeer duidelijk het tegelijkertijd verwerken van instructies van een programma en in- en uitvoer van informatie.

programma



Figuur 3.

De zin van deze konstruktie wordt toegelicht aan het volgende voorbeeld.

VOORBEELD

Stel dat een programma een hoeveelheid informatie (bijv. 100 woorden) inleest, waarmee gerekend gaat worden. Bovendien

moeten nog enige voorbereidingen getroffen worden voordat de berekening met de ingelezen informatie kan plaats vinden. Het programma moet nu zo ingericht worden, dat tijdens het doorlopen van de instructies die de voorbereidingen treffen, de invoer plaats vindt.

Schema van het programma.

Startopdracht voor het lezen
van een blok van 100 woorden

} voorbereidingen

Hier begint de verwerking
van het blok in het geheugen

Vorbereiding en inlezen van het blok geschiedt gelijktijdig.

Wordt het programma opgezet zoals boven geschetst, dan ontstaan er moeilijkheden. Het is dan namelijk niet bekend of het gehele blok van de magneetband al in het geheugen gebracht is als het programma dynamisch op het punt aangekomen is, waar de verwerking van het ingelezen blok begint.

Daarom bestaat er een instructie die in staat is om af te vragen :

Is het gevraagde blok van de magneetband reeds ingelezen in het geheugen? Dus : heeft de startopdracht zijn werk vervuld? Blijkt dat niet zo te zijn, dan wordt het programma op dat punt vastgehouden, totdat de melding komt : Blok inlezen.

Pas dan kan het programma verder gaan.

Schema van het programma

Startopdracht voor het lezen van een blok

} voorbereidingen

Is de startopdracht voltooid?(is het blok ingelezen?)
Begin verwerking van het blok in het geheugen.

Bij een startopdracht behoort dus een afvraagopdracht die informeert of de startopdracht z'n werk voltooid heeft.

Er zijn bij de SERA meerdere magneetbandapparaten die in een

programma gebruikt kunnen worden. Deze magneetbandapparaten zijn aangesloten op kanalen. Nu hebben we gezien dat na een startopdracht voor een magneetbandapparaat de regie weer terug gegeven wordt aan het programma en de volgende opdrachten worden uitgevoerd. Neem aan dat in een programma twee magneetbanden nodig zijn. Ze zijn door omstandigheden beide aangesloten op hetzelfde kanaal.

Stel er wordt een startopdracht voor magneetbandapparaat nr. 1, op kanaal 1 gegeven en iets verderop in het programma een startopdracht voor MB nr. 2 op kanaal 1. MB 1 verwerkt nu zijn startopdracht, maar MB 2 kan niets doen, omdat kanaal 1 bezet is.

Startopdracht voor MB 1 op kanaal 1

Startopdracht voor MB 2 op kanaal 1

Testopdracht of MB 1 voltooid is

Testopdracht of MB 2 voltooid is

Toch wordt na elke startopdracht teruggesprongen naar het programma. Ook in het geval van een startopdracht die niet uitgevoerd kan worden, bijv. omdat het kanaal bezet is. In dat geval wordt deze aanvraag om een apparaat te starten zolang ergens bewaard totdat de mogelijkheid komt om hem uit te voeren.

Als er in een programma dus gebruik gemaakt wordt van meerdere magneetbandapparaten, dan is het wel duidelijk dat er op de een of andere manier een administratie moet worden bijgehouden die aangeeft welke startopdrachten uitgevoerd zijn en welke startopdrachten nog uitgevoerd moeten worden.

Deze hele administratie wordt programmatisch behandeld door een programma dat in het dode geheugen aanwezig is en dat wij zullen noemen : MBLS hetgeen een afkorting is van MAGNEETBAND LEES- EN SCHRIJFPROGRAMMA.

MBLS coördineert de uitvoering van de magneetbandopdrachten en wordt dan ook wel coördinatorprogramma genoemd.

De magneetbandopdrachten die we hier zullen gaan hanteren zijn ook geen echte opdrachten maar pseudo-opdrachten. Het zijn eigenlijk allemaal subroutinesprongen in het pro-

gramma MBLS.

De echte startopdrachten voor de magneetbandapparaten mogen door de programmeur niet gebruikt worden. Ze moeten er uiteraard wel zijn en worden dan ook toegepast in het programma MBLS.

SAMENVATTING

Het programma MBLS regelt voor ons de in- en uitvoer met magneetband op een dusdanige wijze dat parallel gewerkt kan worden. De hele zorg van de administratie is overgedragen aan dit programma. De programmeur heeft niet de beschikking over de echte startopdrachten voor de apparaten maar programmeert met pseudo-instructies. Dit zijn in wezen subroutinesprongen in MBLS en hebben het karakter van "verzoeken".

MBLS regelt deze verzoeken en daarna wordt altijd direkt teruggesprongen naar de opdracht volgend op de pseudo-startopdracht. Hoe MBLS deze verzoeken regelt zal later nader worden besproken.

3.4 MAGNEETBANDOPDRACHTEN.

We zullen nu eerst meer aandacht gaan besteden aan het gebruik van de pseudostartopdrachten en de wijze van toepassing voor de parallelwerking.

Allereerst worden hier de pseudo-opdrachten genoemd die in SERA te gebruiken zijn. Toepassing daarvan volgt daarna.

MVL n Magneetband Vooruit Lezen.

n is het beginadres van het verzorgingsblok. Hier zal later over gesproken worden.

Er wordt een verzoek gedaan aan MBLS om een blok woorden van de magneetband te lezen terwijl de magneetband zich voorwaarts beweegt en dit blok naar een bepaalde plaats naar het geheugen te brengen.

MTL n Magneetband TerugLezen.

Voor n zie MVL.

Er wordt een verzoek gedaan aan MBLS om een blok woorden van de magneetband te lezen.

De magneetband beweegt zich in achterwaartse richting.

MVS n Magneetband Voorwaarts Schrijven.

Voor n zie MVL.

Een verzoek om een blok woorden vanuit het geheugen op de magneetband te schrijven.

Dit is alleen mogelijk als de magneetband vooruit

beweegt.

- MTZ .n Magneetband Terugspoelen Zonder blokkeren.
Voor n zie MVL.
Een verzoek om de magneetband terug te spoelen naar het begin van de band.
- MTM n Magneetband Terugspoelen Met blokkeren.
Voor n zie MVL.
Hier wordt een verzoek gedaan om de magneetband terug te spoelen naar het begin en bovendien wordt het apparaat geblokkeerd zodat het programmatisch niet meer bereikbaar is. Dit blokkeren moet door de operateur gebeuren.
Dit wordt toegepast als de band geschreven is met informatie en daarna door de operateur van het apparaat moet worden afgenomen. De kans dat de band verstoord zou worden, voordat de operateur kan ingrijpen, wordt op deze wijze zeer klein.
- TST n Test of de bijbehorende magneetbandopdracht zijn startopdracht voltooid heeft.
Voor n zie MVL.
Er wordt getest of een blok woorden in- of uitgevoerd zijn. Deze opdracht behoort bij de opdrachten: MVL, MTL, MVS. De opdrachten MTZ en MTM gebruiken geen testopdracht, omdat geen informatie getransporteerd wordt.

De n achter de magneetbandopdrachten verwijst naar een verzorgingsblok van de volgende vorm:

- | | |
|-----|--|
| n | nummer van het magneetbandapparaat |
| n+1 | beginadres in het geheugen van waaruit geschreven of waar naartoe gelezen wordt. |
| n+2 | aantal woorden, de blok lengte, dat getransporteerd wordt. |

Deze volgorde is verplicht bij het verzorgingsblok. Enkele opdrachten maken slechts gebruik van deze informatie die in het adres n staat. Voor spoelen en testen is namelijk het nummer van het magneetbandapparaat slechts van belang. Bij achteruitlezen moet het eindadres van het geheugenblok gegeven worden.

De programmeur nummert zijn apparaten in het programma met 1, 2 en 3 enz. Deze nummers hebben voor hem betekenis. Magneetband 1 is bijv. de oude bestandband, magneetband 2 de nieuwe bestandband en magneetband 3 de mutatieband. Intern in het geheugen van de machine moeten deze magneetbandapparaten echter een nummer krijgen dat aangeeft op welk kanaal en magneetbandapparaat de band gemonteerd is. Nu mag de programmeur dat vanaf zijn bureau nooit bepalen.

Ir. P.A. Tas

Het is de operator die beslist op welk apparaat de band opgespannen zal worden.

Bij het inlezen van het programma door het invoerprogramma, wordt over de bedieningsschrijfmachine gevraagd welk kanaal en apparaatnummer moet worden toegekend aan magneetband 1, 2 enz.

De operator zal dan aan de verschillende magneetbandnummers apparaat- en kanaalnummers toewijzen en dat intypen op de schrijfmachine.

Indien de programmeur verlangt dat zijn banden op magneetbandapparaten gespannen worden die op verschillende kanalen staan, dan moet hij dat op een gebruiksaanwijzing voor de operator vermelden.

VOORBEELDEN.

Hieronder volgen nu enkele voorbeelden van het gebruik van de pseudomagneetbandopdrachten.

	MVL	1030	Lees 100 woorden van het magneetbandapparaat in 3 en breng dit blok in het geheugen te beginnen bij adres 2000.
	TST	1030	Test of apparaat 3 klaar is met transporteren.
1030	3		
1031	2000		
1032	100		
	MTL	PIET	Lees terug van de magneetband (apparaat 2) een blok van 200 woorden en begin dit in te zetten vanaf label BUF. Het geheugen wordt gevuld op adres BUF, BUF-1, BUF-2, ... tot BUF BUF-200.
	TST	PIET	Test of apparaat nr. 2 klaar is met transporteren.
PIET	2		
	SAL	BUF	
	200		

Ir. P.A. Tas

Zoals U ziet is BUF hier het eindadres van het invoergebied. Bij teruglezen wordt het gebied in tegengestelde richting gevuld als bij het vooruitlezen. Het woord zelf komt in de juiste vorm in het geheugen te staan.

JAN	MVS	JAN	Schrijf op de magneetband (app.1) een blok van 100 woorden vanuit het geheugen te beginnen op adres BUF
	TST	JAN	Test of het apparaat klaar is met transporteren.
	1 SAL	BUF	
	100 MTM	JAN	Spoel magneetband 1 terug met blokkeren.

4. EEN AANTAL VOORBEELDEN

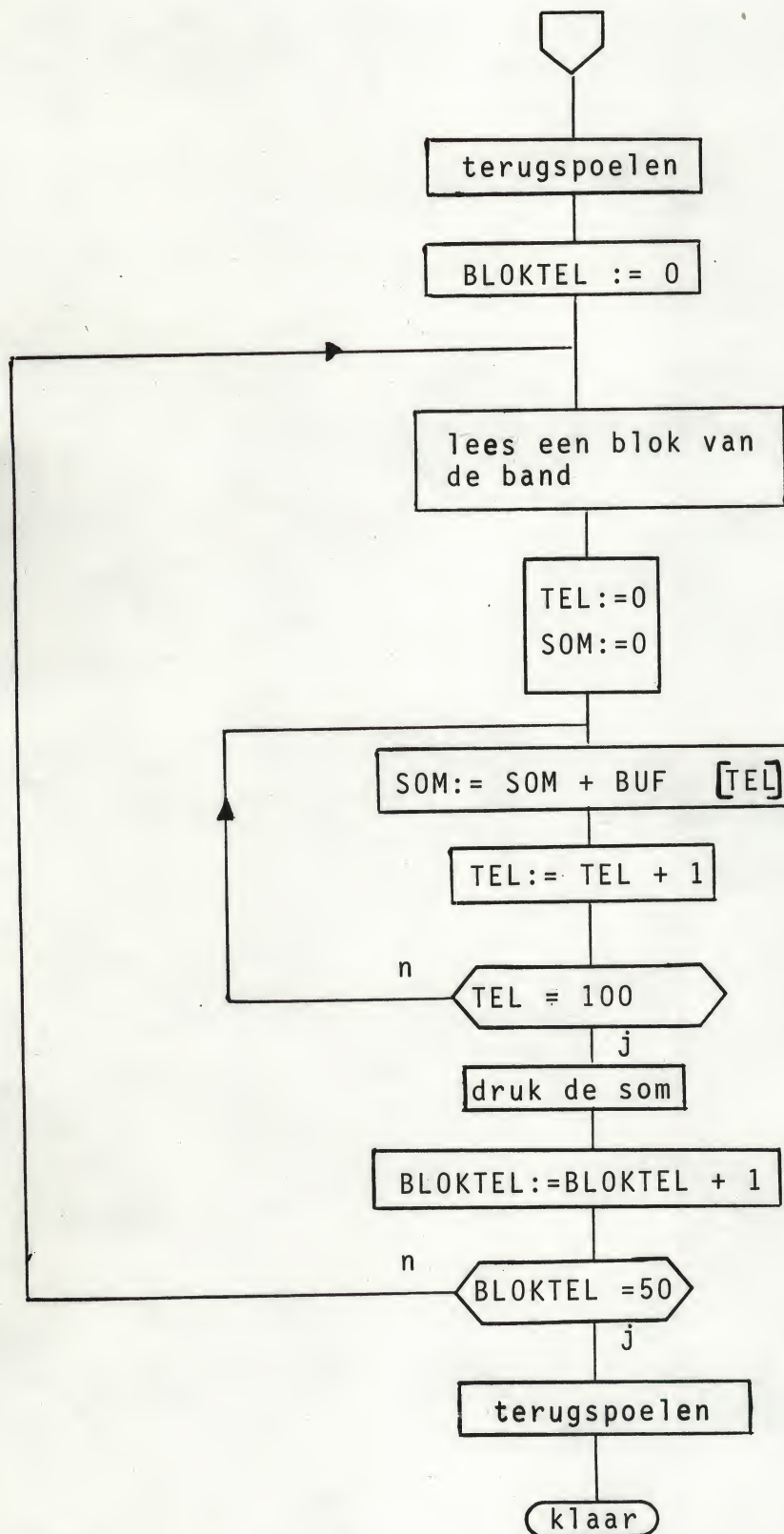
Voorbeeld 1.

Op een magneetband staan 50 blokken van elk 100 woorden. In ieder woord staat een getal. Op elk blok moet eenzelfde soort bewerking uitgevoerd worden, n.l. het optellen van de 100 getallen en het afdrukken van de som op de sneldrukker.

Aangezien er gewerkt wordt met blokken van 100 woorden moet er in het geheugen een ruimte van 100 woorden gereserveerd worden waarin elke keer een blok vanaf de magneetband ingelezen kan worden. Met de informatie in die geheugenruimte kan dan de optelling uitgevoerd worden.

Het eerste woord van dit stuk geheugen krijgt de label BUF.

Het stroomdiagram : (zie volgende pagina)



Dit stroomdiagram heeft weinig toelichting. Een blokteller telt het aantal blokken dat ingelezen is. Elke keer als er 100 getallen opgeteld zijn, wordt de som gedrukt en een nieuw blok wordt in dezelfde geheugenruimte ingelezen. Dit gaat zo door totdat de blokteller gelijk aan 50 is.

STOP	BGN		
	STP	STOP+1	
	BUS	1:120	
	MTZ	V	terugspoelen
	HPA *		
	BPA	BLOKTEL	BLOKTEL :+0
	MVL	V	lees een blok van magneetbandapparaat 1
	TST	V	
	HPA *		
	BPA	TEL	TEL:= SOM :=0
OPTEL	BPA	SOM	
	HPA	SOM	
	MOD	TEL	SOM:=SOM+BUF [TEL]
	OPA	BUF	
	BPA	SOM	
	HPB	TEL	
	OPB *	18	TEL:=TEL+1
	BPB	TEL	
	AFB *	100	
	SNB	OPTEL	
	HPB	SOM	
	KGA	ts12	
	BAB	10:21	druk de som
	DRU		
	HPA	BLOKTEL	
	OPA *	1	BLOKTEL:=BLOKTEL+1
	BPA	BLOKTEL	
	AFA *	50	BLOKTEL+50 ?
BLOKTEL TEL SOM BUF	SNA	OPTEL-4	
	MTZ	V	terugspoelen
	SAL	STOP	
	1		
	SAL	BUF	} verzorgingsblok
	100		
	0		
	0		
	0		
	BGN	BUF+100	
	SRT		

Voorbeeld 2.

Een aantal bladzijden terug werd een voorbeeld van een voor-

raadbestand behandeld.

Het ging over ongeveer 50.000 artikelen. Elk artikel is ondergebracht in een record van 10 woorden groot, waarbij het eerste woord het artikelnummer bevat, het tweede woord het aantal aanwezige artikelen in voorraad. De overige woorden doen hier niets ter zake. De band wordt afgesloten door een sluitblok met in het eerste woord 8 maal de letter z.

Er wordt gemuteerd met kaarten.

In kolom 10 t/m 14 staat het artikelnummer, in kolom 5 de mutatiecode en in kolom 30 t/m 36 de voorraadverandering. Het stroomdiagram van dit probleem is reeds gemaakt en besproken.

Hieronder volgt nu het bijbehorende programma :

Alvorens dit op te schrijven zullen een aantal namen verklaard worden.

KAARTNR een geheugenplaats waarin het art.nummer van de kaart wordt opgeborgen.

VERZI de naam van het verzorgingsblok voor het oude hoofdbestand

VERZ2 de naam van het verzorgingsblok voor het nieuwe hoofdbestand

BLOK de geheugenruimte waarin een blok gelezen wordt of waar vandaan een blok geschreven wordt

Z een woord met sluittekens voor de kaart

EINDTEKEN een woord met sluittekens voor de magneetband.

Aangezien hier twee verschillende magneetbandapparaten gebruikt worden is het ook noodzakelijk twee verzorgingsblokken te schrijven.

Deze verzorgingsblokken verschillen overigens alleen maar in het eerste woord omdat de apparaatnummers niet hetzelfde zijn; het beginadres in beide blokken is gelijk en dit geldt ook voor de bloklengte.

De sluitkaart heeft 5 maal de letter z in de kolommen 10 t/m 14.

De artikelnummers in de records staan in getalvorm.

LEESK	BGN		
	STP	LEESK	
	LSK		lees een kaart
	HAB	10:14	
	HPA	Z	sluitkaart?

	SAB	AFWIKKEL	
	KAG	5	
LEESR	BPB	KAARTEN	breng art.nr. van de kaart weg
	MVL	VERZ1	
	TST	VERZ1	
	HPB	KAARTNR	
	AFB	BLOK	
	SOB	DOOR	
	SAL	LEESR	
DOOR	HAB	5:5	mutatiecode 0?
	SOB	VERDER	
	SAL	LEESK	
VERDER	HAB	30:36	
	KAG	7	tel bij de voorraad de verandering op
	OPB	BLOK+1	
	BPB	BLOK+1	
	MVS	VERZ2	schrijf record weg
	TST	VERZ2	
	SAL	LEESK	
AFWIKKEL	MVL	VERZ1	lees een blok
	HPB	BLOK	
	HPA	EINDTEKEN	laatste blok
	SAB	LAATST	
	MVS	VERZ2	schrijf een blok
	TST	VERZ2	
	SAL	AFWIKKEL	
LAATST	MVS	VERZ2	
	TST	VERZ2	
	SAL	LEESK-1	
Z	"ZZZZZ"		
KAARTNR	0		
EINDTEKEN	"ZZZZZZZZ"		
VERZ1	1		
	SAL	BLOK	
	10		
VERZ2	2		
	SAL	BLOK	
	10		
BLOK	BGN	BLOK+10	
	SRT	LEESK-1	

5. ENIGE VOORBEELDEN MET PARALLELWERKING

Tot nu toe is in de voorbeelden geen gebruik gemaakt van parallelwerking.

Voorbeeld 1.

Op een magneetband staan 50 blokken van elk 100 woorden.

Op elk blok moet dezelfde soort bewerking uitgevoerd worden. Deze bewerking zal hier niet geprogrammeerd worden en zal in het stroomdiagram en in het programma symbolisch worden aangeduid met BEWERKING. Aangezien er gewerkt wordt met blokken van 100 woorden moet er in het geheugen een ruimte van 100 woorden gereserveerd worden, waarin elke keer een blok vanaf de magneetband ingelezen kan worden. Het eerste woord van dit stuk geheugen zal de label BUF krijgen.



Na enige voorbereidende opdrachten wordt een startopdracht gegeven voor het lezen van een blok in het geheugen. Daarna gaat het programma direkt verder met de opdrachten die TELLER verhogen.

Als dat gebeurd is kan de bewerking op het ingelezen blok beginnen. Alvorens dat te doen wordt getest of de startopdracht voltooid is, dus of het blok inderdaad ingelezen is. Zo nee, dan wordt gewacht totdat er een sein komt dat het apparaat een blok heeft verplaatst en daarna wordt de be-

werking uitgevoerd.

Zo ja, dan kan de bewerking direkt beginnen.

Het programma:

	BGN		
	BSA	TELLER	0 → TELLER
	BPA	TELLER	
	MTZ	VERZB	terugspoelen
LEES	MVL	VERZB	lees een blok
	HPA	TELLER	
	OPA *	1	TELLER:=TELLER+1
	BPA	TELLER	
	TST	VERZB	
	BEWERKING		
	HPA	TELLER	
	AFA *	50	is TELLER=50?
	SNA	LEES	
	STP		
TELLER	0		
BUF	BGN	BUF+100	
VERZB	1		
	SAL	BUF	
	100		
	SRT	TELLER-1	

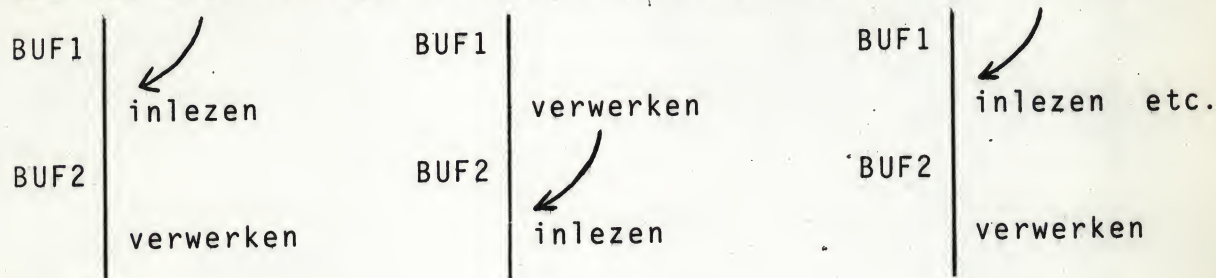
Er is bij bovenstaande wijze van programmeren nauwelijks gebruik gemaakt van de mogelijkheid parallel te werken. Slechts het verhogen van de TELLER gaat tegelijkertijd met het inlezen van een blok.

Het probleem is recht toe recht aan gecodeerd.

Het zou prettig zijn de BEWERKING gelijktijdig te laten verlopen met de invoer van een nieuw blok met gegevens. Daarom wordt een tweede stuk geheugen gereserveerd. De stukken geheugen worden in dit geval ook wel buffers genoemd.

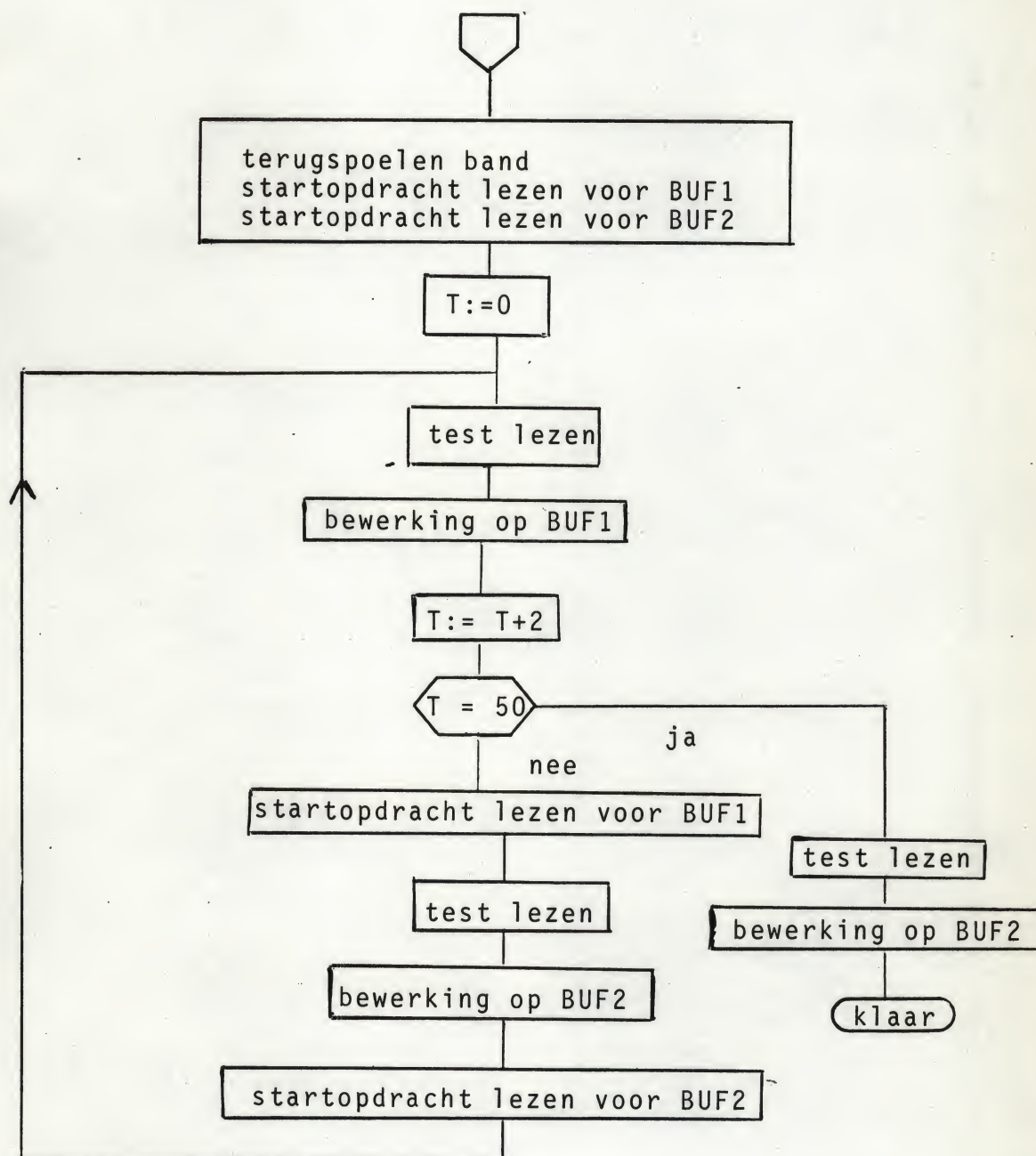
De buffers krijgen de labels BUF1 en BUF2.

Terwijl bijv. in BUF1 informatie ingevoerd wordt, worden de gegevens in BUF2 verwerkt. Is dit laatste klaar, dan wordt getest of BUF1 al gevuld is. Zo ja, dan worden de gegevens in BUF1 verwerkt en tegelijkertijd BUF2 weer gevuld. Zo nee, dan wacht het programma tot het sein komt dat het blok is ingelezen, waarna verder gerekend kan worden.



Terwijl de ene buffer wordt verwerkt, wordt de andere volgelezen.

Het nu volgende stroomdiagram toont een methode waarop dit geprogrammeerd zou kunnen worden.



Bespreking van het stroomdiagram.

Er worden 2 startopdrachten voor het lezen van de magneetband gegeven. De eerste startopdracht kan direkt uitgevoerd worden, de tweede geldt voor hetzelfde magneetbandapparaat en wordt zolang opgeborgen in MBLS en uitgevoerd als de eerste klaar is. De eerste opdracht leest in buffer 1, de tweede in buffer 2.

Dan wordt een teller op 0 gezet. Nu wordt gevraagd of de magneetband al een blok heeft ingelezen. Als dat zo is dan moet dat die voor buffer 1 zijn en kan het programma vervolgd worden. Anders blijft het programma in de wachttoestand totdat gemeld wordt : Er is een blok van die magneetband ingelezen. Als het programma zijn weg kan vervolgen wordt buffer 1 verwerkt, tegelijkertijd zal nu buffer 2 gevuld worden want de startopdracht die daar vroeger voor gegeven is en die zolang was opgeslagen in MBLS kan nu worden uitgevoerd. *Dit is duidelijk parallelwerking.*

Als buffer 1 verwerkt is wordt de teller met 2 verhoogd om te kijken of er al 50 blokken zijn geweest. Is dat niet zo dan wordt eerst een startopdracht lezen voor buffer 1 gegeven omdat we klaar zijn met rekenen in die buffer. Direkt daarna vervolgt het programma zijn weg.

We zouden nu graag buffer 2 gaan verwerken maar daartoe moet eerst getest worden of die al ingelezen is. Denk eraan dat een testopdracht slechts betrekking heeft op een apparaat op een kanaal en niet geïnteresseerd is waar de informatie naar toe gaat of waar die vandaan komt.

Als inderdaad het blok is ingelezen wordt de bewerking uitgevoerd, daarna deze buffer weer gevuld en door het terugspringen wordt weer getest of er al weer een blok ingelezen is.

Is de teller gelijk aan 50 dan moet er nog een blok verwerkt worden. Daarom moet getest worden of dit blok al is ingelezen in buffer 2 en zo ja dan vindt de bewerking plaats waarna het programma klaar is.

Het programma:

LUS	BGN		
	MTZ	VERZ1	terugspoelen
	MVL	VERZ1	startopdracht BUF1
	MVL	VERZ2	startopdracht BUF2
	HPA *		
	BPA	T	T:=0
	TST	VERZ1	BUF1 volgelezen?
	BEWERKING	BUF1	
	HPA	T	
	OPA *	2	T:=T+2
	BPA	T	
	AFA *	50	T = 50

	SOA	EIND	
	MVL	VERZ1	startopdracht lezen voor BUF1
	TST	VERZ2	BUF2 volgelezen?
	BEWERKING	BUF2	
	MVL	VERZ2	startopdracht lezen voor BUF2
	SAL	LUS	
EIND	TST	VERZ1	BUF2 volgelezen?
	BEWERKING	BUF2	
	STP		
T	0		
VERZ1	1		
	SAL	BUF1	
	100		
VERZ2	1		
	SAL	BUF2	
	100		
BUF1	BGN	BUF1+100	
BUF2	BGN	BUF2+100	
	SRT	EIND+2	

Voorbeeld 2

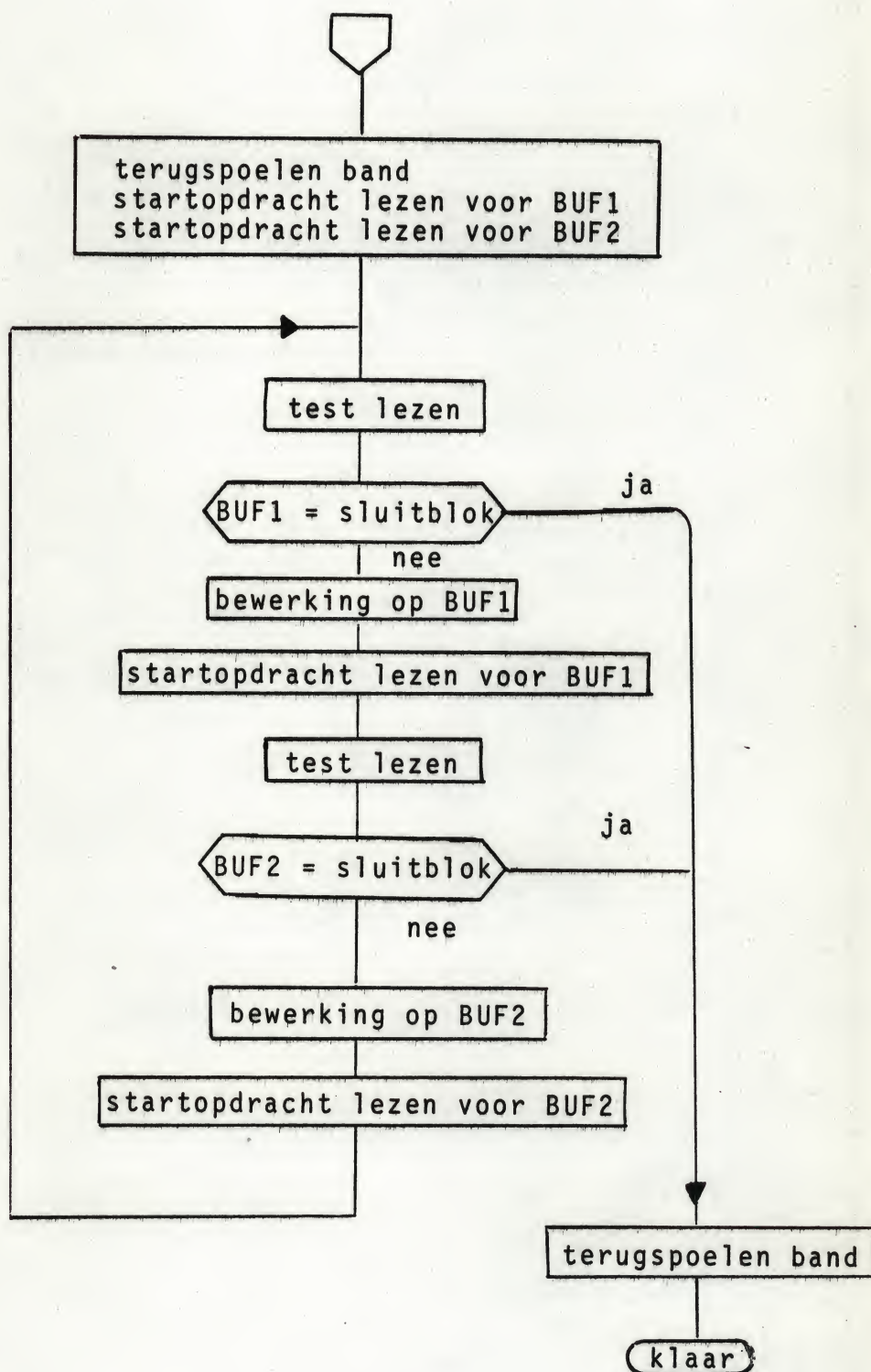
Vaak is het aantal blokken op magneetband onbekend en wordt daarom de informatie op de band afgesloten met twee sluitblokken van dezelfde woordlengte als de informatieblokken. Zo'n sluitblok bevat dan een eindteken, bijv. in het eerste woord van het blok het getal +999999999999.

Elke keer dat een blok is ingelezen moet nu getest worden of dit een sluitblok is.

Er moeten 2 sluitblokken zijn, omdat tijdens het testen op het eindteken van een blok in parallelwerking een volgend blok wordt ingelezen.

Zouden er geen 2 sluitblokken op de band staan dan kan een magneetbandleesfout optreden omdat het magneetbandapparaat een blok tracht in te lezen dat er niet is. De informatie in dit laatste blok is eigenlijk oninteressant als er maar op de band een blok staat met de juiste bloklengte.

Het vorige voorbeeld wordt nu weer geprogrammeerd, alleen nu wordt aangenomen dat het aantal blokken onbekend is en dat de informatie wordt afgesloten met twee blokken waarin het eerste woord 8 maal de hexade z staat.



	BGN		
	MTZ	VERZ1	terugspoelen
	MVL	VERZ1	start voor BUF1
	MVL	VERZ2	start voor BUF2
WEER	TST	VERZ1	
	HPB	BUF1	
	HPA	Z	sluitblok?
	SAB	---KLAAR	
		BEWERKING OP BUF1	
	MVL	VERZ1	start voor BUF1
	TST	VERZ1	
	HPA	Z	sluitblok?
	HPB	BUF2	
	SAB	---KLAAR	
		BEWERKING OP BUF2	
	MVL	VERZ2	start voor BUF2
	SAL	---WEER	
KLAAR	MTZ	VERZ1	
	STP		
Z		"zzzzzzzzz"	
VERZ1	1		
	SAL	BUF1	
	100		
VERZ2	1		
	SAL	BUF2	
	100		
BUF1	BGN	BUF1+100	
BUF2	BGN	BUF2+100	
	SRT	KLAAR	

Na het terugspoelen worden er twee startopdrachten lezen gegeven voor BUF1 en BUF2. De daaropvolgende testopdracht vraagt of er een blokeinde-melding is geweest. Zodra BUF1 nu gevuld is wordt gekeken of het eerste woord gevuld is met de letter z.

Zo nee, dan wordt de bewerking uitgevoerd, daarna een startopdracht voor BUF1 gegeven en gekeken of het apparaat al weer een blokeinde-melding heeft gegeven, hetgeen zou betekenen dat BUF2 volgelezen is.

Na de test op het sluitblok, wordt ook dit blok verwerkt en er wordt weer een startopdracht lezen voor BUF2 gegeven. Dan volgt een terugsprong naar de TST opdracht die weer de blokeinde-melding test. enzovoorts.

7. BUFFERTECHNIEK

7.0 INLEIDING

Vaak komen administratieprogramma's voor, die tot doel heb-

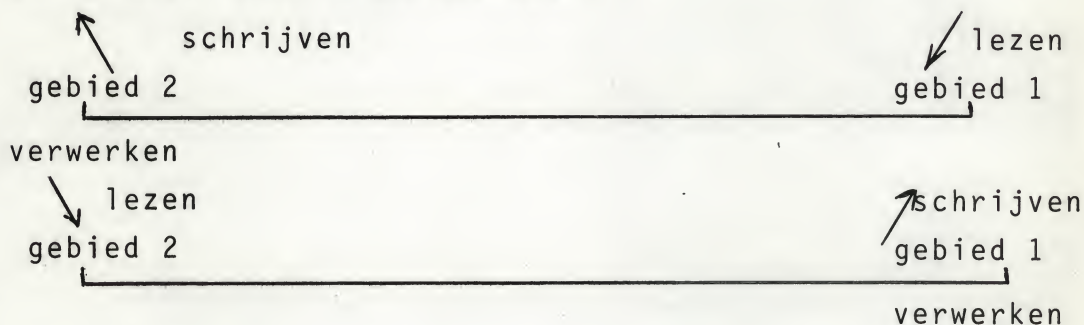
ben een nieuw bestand te maken door het oude bestand met mutaties bij te werken. Het oude bestand staat op een magneetband. De mutaties staan op kaarten of een magneetband en er is een band aanwezig voor het nieuwe bestand. Als verondersteld wordt dat de mutaties in het geheugen zijn opgenomen ziet het vereenvoudigde probleem er als volgt uit :

Lees van band 1, muteer indien noodzakelijk en schrijf op band 2. Het ligt voor de hand dat ook hier gezocht moet worden naar een oplossing die gebruik maakt van parallelwerking.

Er zijn meerdere methoden die toegepast kunnen worden. Aan de hand van een schets van de in- en uitvoergebieden zullen we die hier bespreken.

We zullen aannemen dat er een magneetband is waarop het oude bestand staat, een magneetband voor het nieuwe bestand en dat de mutaties op kaarten staan.

De woorden "lezen" en "schrijven" hebben dan betrekking op de twee magneetbanden. De magneetbanden zijn op verschillende kanalen aangesloten.

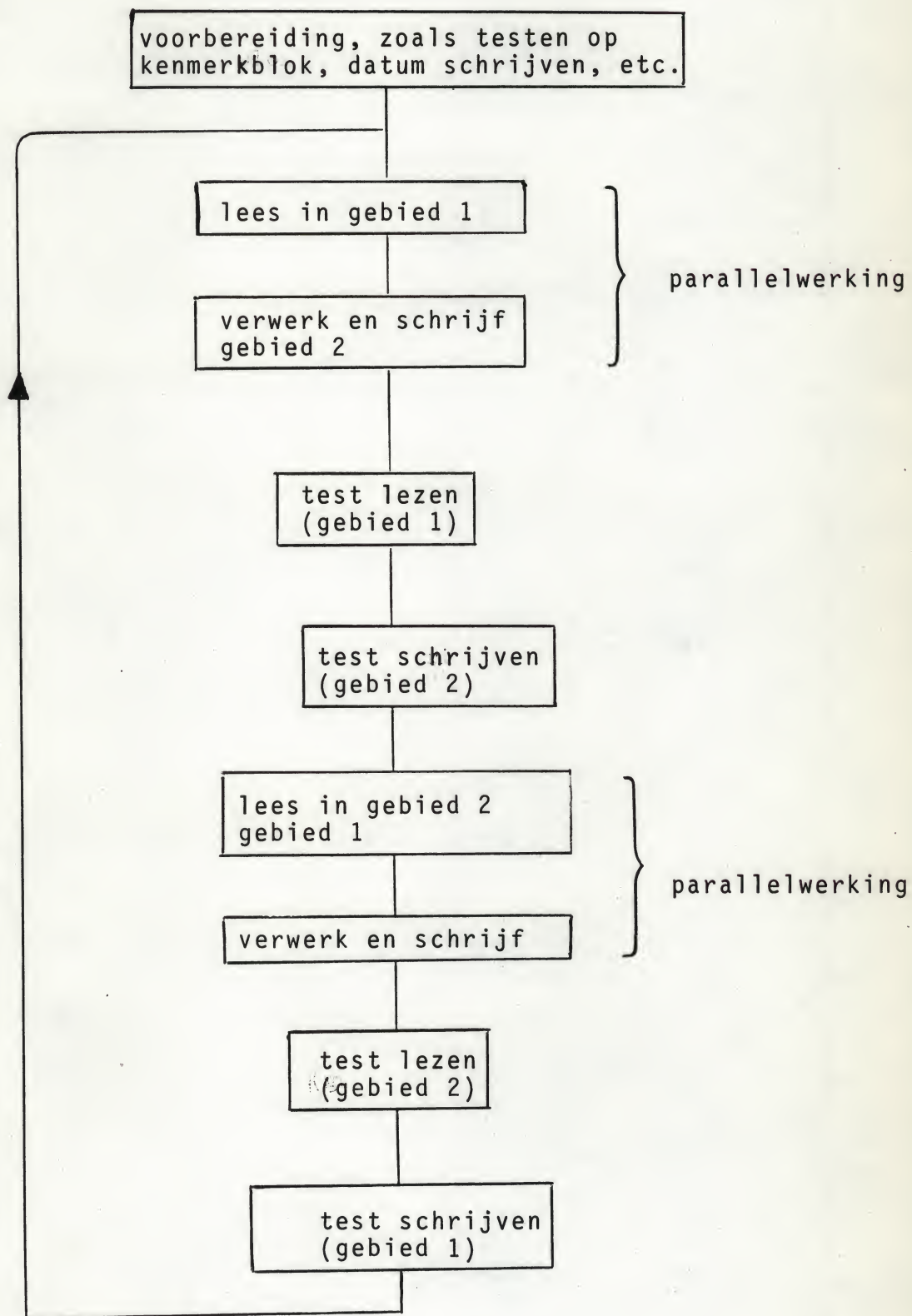


Deze methode is reeds bekend. Afwisselend wordt in het ene gebied gelezen, terwijl het andere gebied verwerkt en daarna weggeschreven wordt.

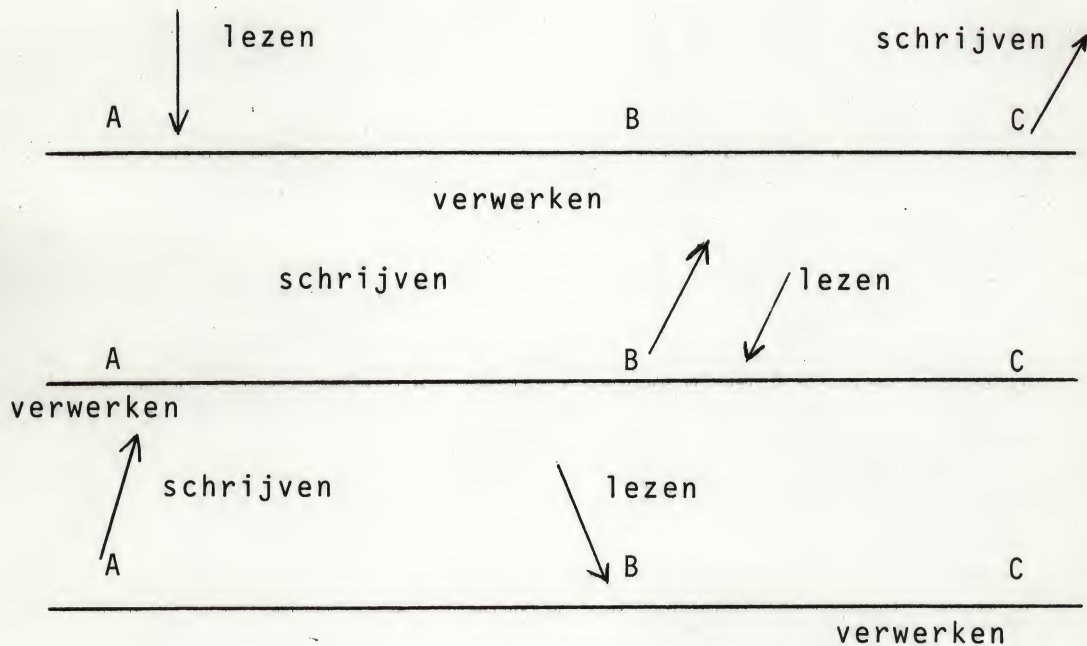
Er wordt parallel gewerkt, maar er wordt niet maximaal gebruik gemaakt van de mogelijkheden. Het vervelende is n.l. dat parallel aan het lezen van een blok een verwerking en het schrijven van een blok moet plaatsvinden.

Verwerken en schrijven neemt meer tijd dan alleen lezen. Er zullen wachttijden optreden die liggen in de orde van de verwerkingstijd. Er moet getracht worden deze wachttijd ook op te vangen.

Het principeschema ziet er als volgt uit :

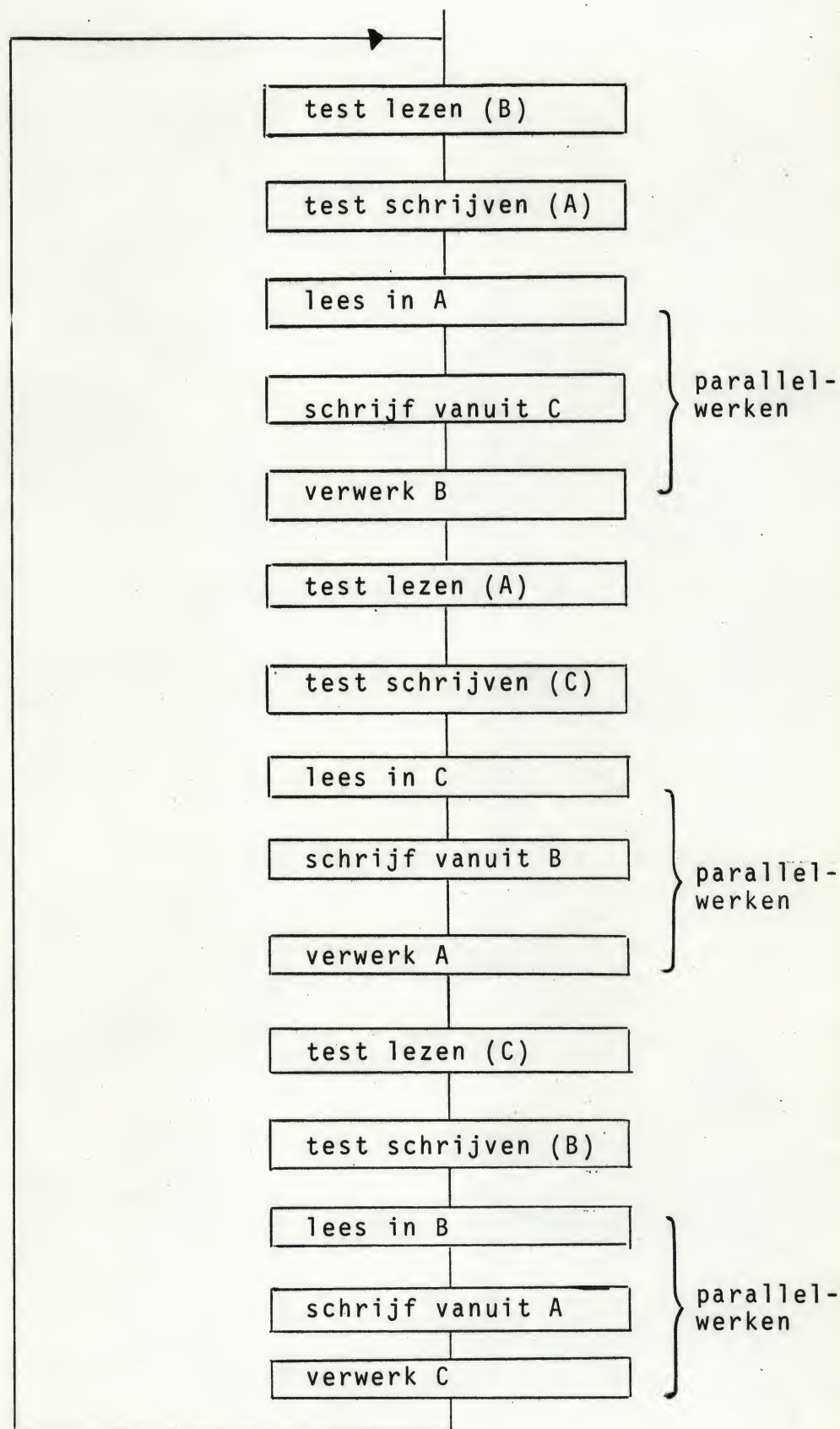


7.2 DRIEBUFFERSYSTEEM



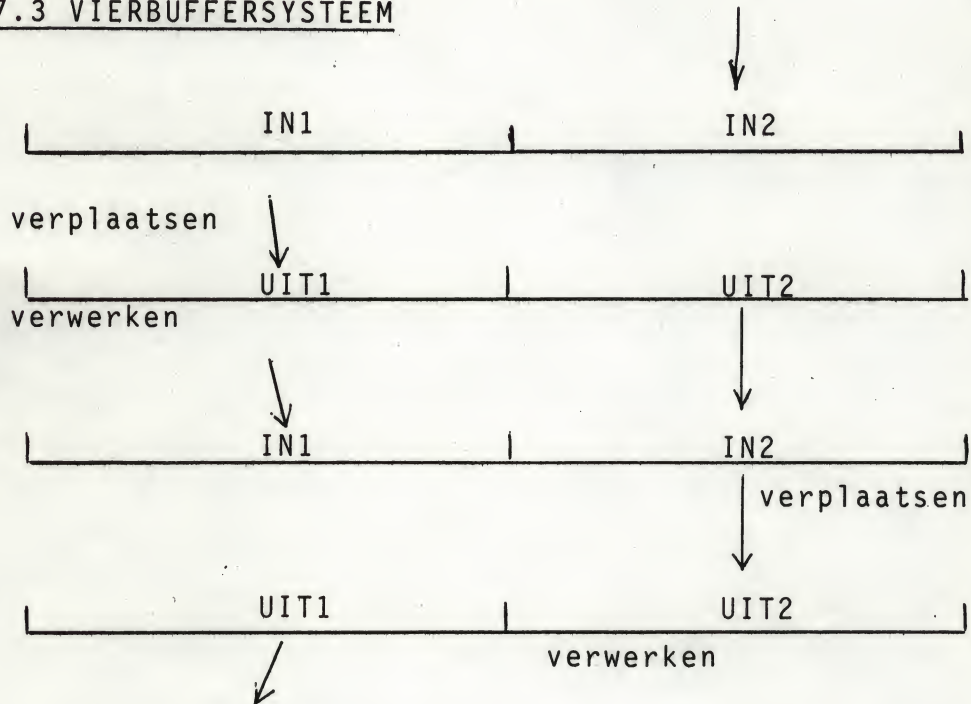
Terwijl in A een blok gelezen wordt, vindt tegelijkertijd verwerking plaats van informatie in B en wordt vanuit C een blok weggeschreven. In de daaropvolgende situatie wordt A verwerkt, B weggeschreven en C in een blok gelezen. De derde situatie is dat A weggeschreven, in B een blok gelezen en C verwerkt wordt. Er kunnen nu nog slechts wachttijden gaan optreden indien de verwerking langer gaat duren dan het lezen of schrijven van een blok.

Het programmaschema. (zie volgende pagina)



De programmering van een driebuffersysteem is vrij ingewikkeld. Als laatste methode wordt een vierbuffersysteem besproken.

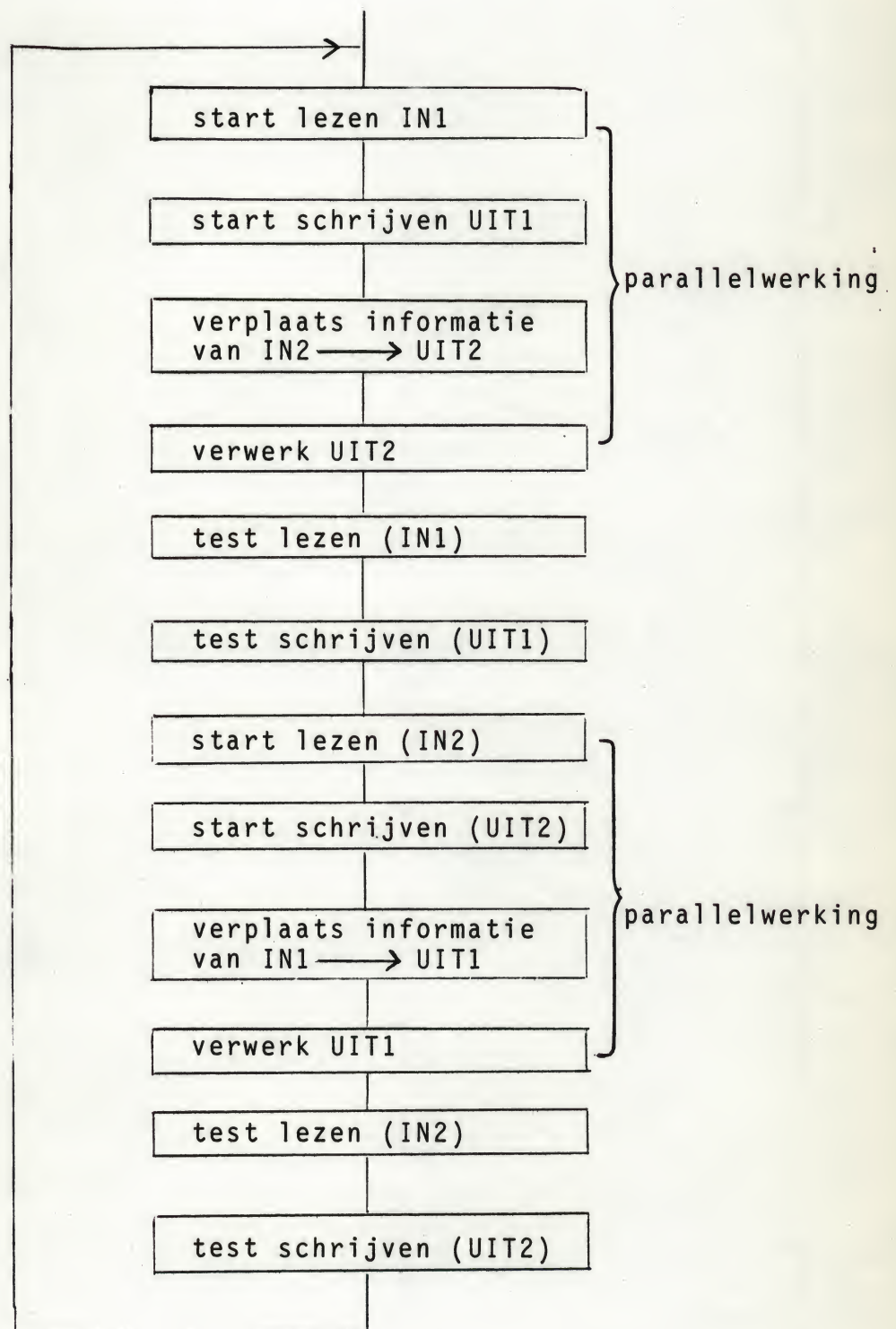
7.3 VIERBUFFERSYSTEEM



Dit systeem maakt gebruik van 2 invoergebieden en 2 uitvoergebieden. Terwijl informatie verplaatst wordt van IN1 naar UIT1 en daarna verwerkt wordt, wordt een blok gelezen in IN2 en een blok geschreven vanuit UIT2.

Door de volgende keer te lezen in IN1 en te schrijven uit UIT1, en de verwerking te laten plaatsvinden in UIT2, ontstaat parallelwerking. Het vierbuffersysteem is eenvoudiger te programmeren dan het driebuffersysteem.

Nadelig is natuurlijk dat er vier in plaats van drie buffers gebruikt worden, terwijl bovendien een verplaatsing van informatie optreedt. Dit laatste kan weer tot wachttijden aanleiding geven indien verwerking zelf al te veel tijd in beslag neemt en zodoende de totale tijd (verwerkings-+verplaatsingstijd) de schrijf- of leestijd van een blok gaat overschrijden.



8. PROGRAMMERINGSTECHNIEKEN

8.0 DE SCHRIJFRING

De zich op de magneetband bevindende informatie gaat door schoonmaken of door opnieuw beschrijven van de magneetband verloren. De magneetbandapparaten zijn daarom voorzien van een blokkeringsinrichting, om te voorkomen dat per vergissing informatie overgeschreven wordt. De blokkering wordt opgeheven door op de band een ring te klemmen.

In de administratie komt het geregeld voor dat een programma één of meerdere magneetbanden, geheel volbeschreven met gegevens, te verwerken krijgt. Dit stelt enige eisen aan de vorm waarin de gegevens op band staan.

8.1 KENBLOK

Het zal noodzakelijk zijn om voorzorgsmaatregelen in het programma te nemen om te verhinderen dat tijdens de afloop van het programma een foutief opgespannen band gelezen wordt.

Daarom bevat het eerste blok van een magneetband, het kenblok, informatie die de band kunnen identificeren.

Bijvoorbeeld :

- a. naam*
- b. nummer van de band. Een proces kan gebruik maken van meerdere banden.*
- c. datum waarop de band gemaakt is.*
- d. datum tot wanneer de inhoud van de band bewaard moet blijven.*

8.2 SLUITBLOKKEN

Het is beter de band niet geheel tot het eind te beschrijven, maar dicht bij het einde de informatie af te sluiten met twee sluitblokken. Het programma wordt op deze manier gewaarschuwd dat het einde van de band bereikt is.

De informatie in zo'n sluitblok moet, voor het programma herkenbare, sluittekens bevatten.

In vele gevallen bevat dit blok nog informatie over de inhoud van de band, zoals : Aantal blokken, "hash total", telling etc.

8.3 GROOTVADERSYSTEEM

Bij het verwerken van administratieve gegevens, wordt in vele gevallen een invoer- en een uitvoerband gebruikt waarop resp. de oude en de nieuwe informatie staat. De oude in-

formatie heeft dan strikt genomen geen waarde meer. Toch moet de invoerband bewaard blijven omdat door technische storingen of door foutieve behandeling de nieuwe band wel beschadigd kan worden.

Pas wanneer de nieuwe band weer gebruikt wordt als invoerband en er een nieuwe uitvoerband gemaakt is, kan de oude invoerband overgeschreven worden.

Als het kleinkind geboren is wordt de grootvader overbodig.

8.4 HERSTARTEN

Bij het gebruik van veel magneetbanden kan het zinvol zijn, de stand van zaken te fixeren na het vol schrijven van elke band.

Voorbeeld :

Om 10 nieuwe magneetbanden samen te stellen moet er gebruik gemaakt worden van 7 banden A1, A2, A7 en 7 banden B1, B2, B7.

Indien bij het samenstellen van de 5e band een fout optreedt, of het programma moet afgebroken worden, dan zal bij opnieuw starten weer van voren af aan begonnen moeten worden.

Dit kan voorkomen worden door achter het kenblok van iedere nieuwe band de belangrijkste gegevens, noodzakelijk voor een eventuele herstart, in een blok op de band te schrijven.

De gegevens kunnen o.a. zijn : Stukken van het geheugen en accumulatoren.

8.5 DIENSTBAND

Het is mogelijk in plaats van gegevens, volledige programma's op de magneetband te brengen. Door een speciaal dienstprogramma kan het gewenste programma op de magneetband opgezocht worden en het geheugen gelezen worden.

Dit is speciaal zinvol voor de programma's die geregeld gebruikt en op deze manier snel in het geheugen gebracht kunnen worden. De programma's waarom het hier gaat, zijn veel gebruikte hoofd- en subprogramma's zoals : compilers, in- en uitvoerprogramma's, tracers, klantenprogramma's, etc.

9. VOOR- EN NADELEN VAN MAGNEETBANDEN

Voordelen : *Vrij snelle overdracht van informatie van en naar het kerngeheugen.*

Zeer grote capaciteit; zeer veel gegevens kunnen op een band worden opgeslagen.

Goedkoop t.o.v. trommel of schijven geheugen.

Nadelen : De informatie staat in serie op de band.
Vereist zorgvuldige behandeling.
Onderhevig aan slijtage.

10. SAMENVATTING

De magneetband is tot op heden nog de meest gebruikte vorm van extern geheugen. Het is een groot nadeel dat de informatie in serie op de band staat.

In gevallen waarin dit zwaar weegt, verdient toepassing van schijvengeheugens, trommelgeheugens of magnetische kaartgeheugens de voorkeur.

Het gebruik van magneetbanden is te onderscheiden in :

- a. In- en uitvoermedium. Programma's en gegevens worden eerst gekonverteerd naar een magneetband, waarna de magneetband als invoer wordt gebruikt.
Resultaten van programma's worden naar een magneetband gebracht, waarvan de inhoud later op de sneldrukker wordt gedrukt.
- b. Opslagplaats. Bewaren van bestanden. Dienstband.
- c. Tijdelijk geheugen. Tijdens het proces wordt de magneetband als extra geheugen gebruikt.

De magneetbandopdrachten zijn niet moeilijk te hanteren. Het toepassen van de buffertechnieken maakt het programmeren echter gecompliceerd. Het is noodzakelijk dat het aantal startopdrachten minstens even groot zijn als het aantal TST opdrachten.

De coördinator MBLS is bij de huidige machines een zeer ingewikkeld programma.
Meestal zal de coördinator ook de parallelwerking van kaarten, ponsband en sneldrukker verzorgen en dit bovendien toepassen op meerdere programma's die zich in het geheugen bevinden, rekening houdend met een prioriteitsregeling.